

Introduction

In this application note we focus on how the ispClock™5406D and a low-cost CMOS oscillator can be utilized to drive the reference clock for SERDES-based applications. SERDES applications require accurate and low-jitter clock sources, which the ispClock5400D is more than able to provide. We will share the jitter results from bench test data using the LatticeECP3™ FPGA with SERDES and the ispClock5406D in-system-programmable differential clock distribution chip. Both XAU1 and SDI video application frequencies were tested, proven to work successfully in both applications and surpass their jitter requirements. The focus is on the use of a low-cost, readily-available CMOS oscillator, interfaced through the ispClock5406D to generate the clean differential SERDES ref-clock.

Note that although a LatticeECP3 was used for the data gathered in this application note, the ispClock5400D can also be used with other SERDES-capable ICs in a similar manner.

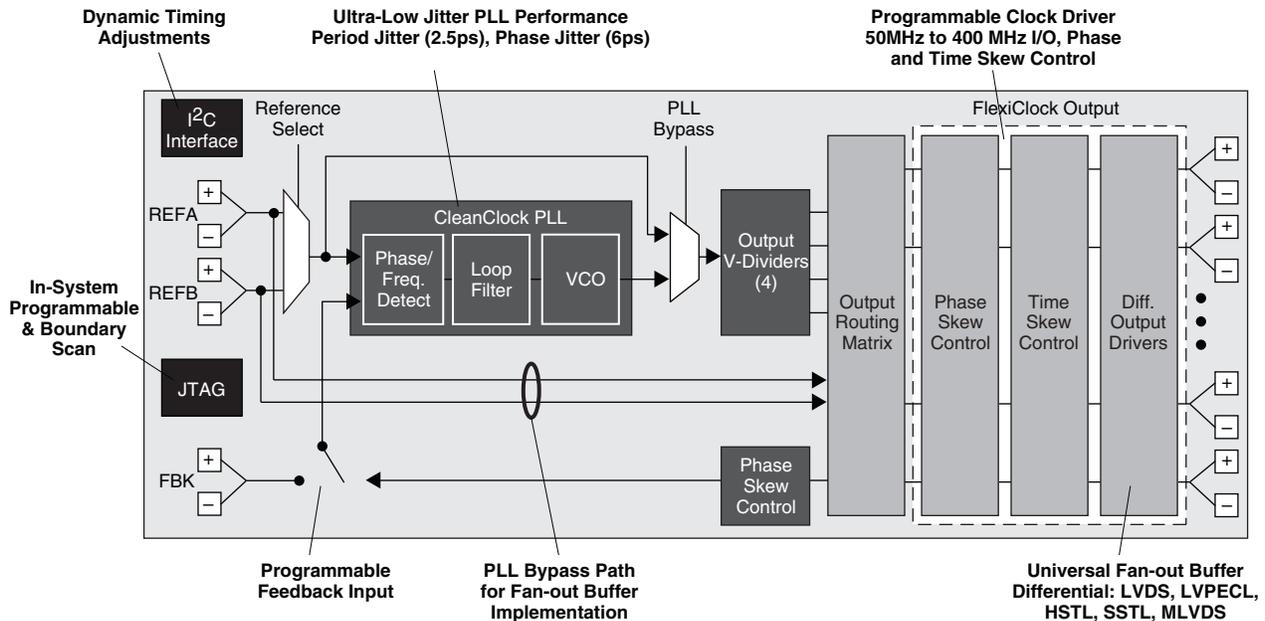
This document complements AN6080, [Using a Low-Cost CMOS Oscillator as a Reference Clock for SERDES Applications](#), which describes the design practices for interfacing an ispClock5400D to a LatticeECP3 FPGA.

The ispClock5400D Family

The ispClock5406D and ispClock5410D are in-system-programmable differential clock distribution ICs designed for use in high-performance communications and computing applications. The ispClock5400D family features the CleanClock™ ultra-low phase noise, third-generation PLL. The FlexiClock™ output section supports multiple logic standards and dual skew control features.

The configuration of each device is held in on-chip non-volatile memory that is reprogrammable through a JTAG interface. Certain aspects of the device can be modified “on-the-fly” via an I²C interface.

Figure 1. ispClock5400 Block Diagram



The ispClock5400D device family features in-system-programmable differential clock distribution with fully programmable features that allow users to program in different frequencies based on dividers and PLL functions. Additional value is added with the multiple differential I/O support for various standards while maintaining the low jitter required for high performance systems. The ispClock5400D family supports differential output drivers for programmable differential input reference/feedback standards: LVDS, LVPECL, HSTL, SSTL, HCSL. Included on-chip are the following features: programmable termination and a clock A/B selection multiplexer, programmable time skew, programmable phase skew and various programmable output enable features. Through I²C the user has access to nearly all the programmable features of the ispClock5400D. The ispClock5400D provides all of this yet still maintains extremely low jitter.

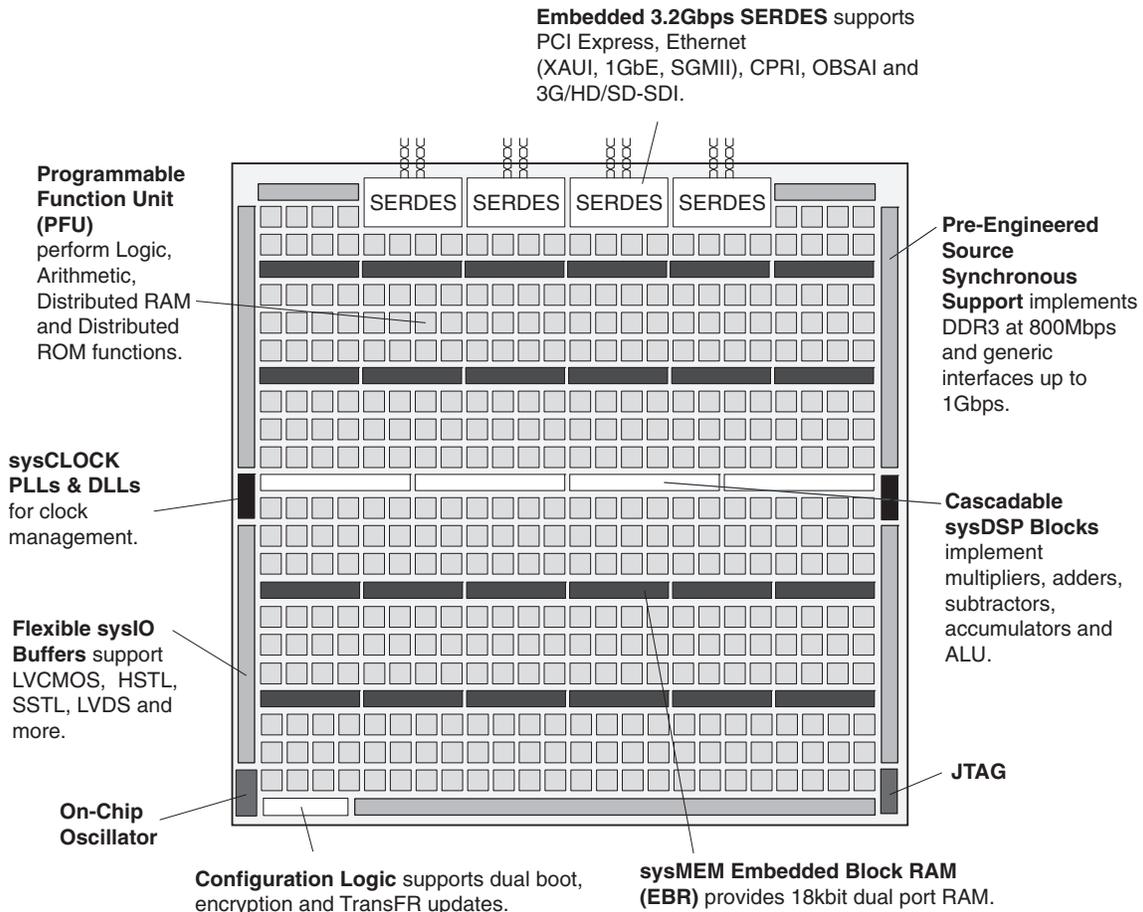
- Ultra low cycle-to-cycle jitter (29ps p-p)
- Ultra low period jitter (2.5ps)
- Low output-to-output skew (<100ps)

Packaging includes the 48-pin and 64-pin QFNS packages.

LatticeECP3 FPGA Family

The LatticeECP3 family is the third-generation mid-range, high-value FPGA from Lattice Semiconductor, which offers the industry's lowest power consumption and price of any SERDES-capable FPGA. The LatticeECP3 offers multi-protocol 3.2G SERDES with XAU1 jitter compliance and SDI video capability, DDR3 memory interfaces, powerful DSP capabilities, high density on-chip memory and up to 149K LUTs, all with half the power consumption and half the price of competitive SERDES-capable FPGAs.

Figure 2. LatticeECP3 Block Diagram



Note: For the SERDES application and testing, an Epson CMOS oscillator was used (part number SG-710ECK 78.125 MHz in a 5x7 package and soldered to the board). SMA connectors and cables were used to drive the reference clock.

SERDES Reference Clock Source Challenges

Typically the challenges for any reference clock for protocols such as XUI, that controls SERDES devices or complex FPGA communications protocols are quite expensive, subject to power supply noise, coupling of high-speed signals and jitter frequency components that are not easily understood or solved.

One method of driving the SERDES reference clock is with a true differential output oscillator purchased at the correct frequency and specifically chosen for low jitter and phase noise. This can cost a great deal more than the solution that is presented here and is not as flexible for changes. The technology of low jitter differential oscillators typically does not include internal PLLs to generate the output frequency: these can have noisy sidebands and multi-modal distributions when analyzed in the frequency domain.

There are also challenges to finding the ideal termination and differential I/O logic standards and maintaining quiet power to the source. These oscillators may run in price from \$12 to \$50 depending on specifications, quantity, packaging, temperature ranges, etc. For these reasons Lattice presents the ispClock5406D as a low-cost flexible solution to driving the SERDES reference clocks.

This is a clean, flexible solution that provides the user a quick-to-market, easy-to-design solution to solving clock distribution problems on FPGA interfaces. Again, the focus is to present the data measured in a real world environment, show the simple interface techniques and summarize the output jitter of the signals at the SERDES level.

Several different setups were tested and the material presented is based on an input frequency from an Epson low-cost CMOS single-ended oscillator running at 78.125MHz. In a XUI application, this is fed into the ispClock5406D and it is configured to output LVPECL at 156.25 MHz. In a XUI application, this is fed into the ispClock5406D and it is configured to output LVPECL at 156.25 MHz or 312.5 MHz. This results in the output of the SERDES running at 312.5 Gbps.

The following section describes how some data reports and manufacturer data sheets report a term referred to as UI, or Unit Interval. These simple relationships are related to the jitter and the amount of jitter when referenced to the period of the waveform.

Jitter Calculations and Forms of Jitter

Differential Programmable Clock Buffer, Jitter calculations: UI, Unit Interval

Looking at the scope plot distributions and data summary at the bottom of each plot, we see the following data parameters that represent the data plots, distributions, etc in each setup.

- Total Jitter: Tj
- Random Jitter: Rj
- Period Jitter: Pj
- Deterministic Jitter: Dj

Conditions:

- Frequency in: 78.125MHz, (drives input reference for ispClock5406D)
- Low-cost CMOS single-ended oscillator as source frequency
- Frequency option 1 (ispClock5406D): 156.25MHz differential LVPECL output drivers to feed the SERDES reference clock

- Frequency option 2 OUT (ispClock5406D): 312.5 MHz Differential LVPECL output drivers to feed the SERDES reference clock
- Internally, the PLL and SERDES from the LatticeECP3 FPGA drive out at 3.12GHz

Using the data from the scope, we can convert these common jitter formats to “UI” form, Unit Interval, or a proportion of the period of the output frequency. This form allows us to look at the data in a format relative to frequency and is commonly used in specifications for high-speed serial data transfer protocols.

Use the following data for this example:

$$T_j = 104.84\text{ps (see Table 1 Epson fixed frequency)}$$

$$\text{Frequency out} = 3.12\text{GHz}$$

$$\text{Period for the output frequency } T_{\text{period}} = 1/\text{Output Frequency}$$

$$T_{\text{period}} = 1/(3.12\text{E9})$$

$$T_{\text{period}} = 3.205\text{E} - 10 = 0.3205\text{E} - 9 \quad 320\text{ps}$$

Jitter in UI is equal to the jitter measured divided by the period of the output frequency

Total Jitter in UI form:

$$T_j = 104.84\text{ps}/320\text{ps}$$

$$T_j = 0.3276 \text{ UI}$$

$$T_j = 0.33 \text{ UI is shown in summary tables for measured data}$$

ispClock5406 in XAUI Based Applications

The data shows jitter analysis of an Epson oscillator with the ispClock5406D with comparison data using the Connor-Winfield Differential Oscillator and the Agilent 8133A clock generator. Data was collected for TX jitter at 3.125Gbps/PRBS7 from the LatticeECP3 and shown in Table 1.

The Epson fixed oscillator with the ispClock5406D combination as a reference clock source to a BERT analyzer ran PRBS7 serial data for more than 24 hours without any errors.

This data includes temperature variations from -55°C to +85°C on the ispClock5406 device.

The second and third groupings are temperature variations and SERDES X10(312.5MHz reference clock) and X20(156.25MHz reference clock) modes.

With the 156.25MHz clock in X20 mode, the XAUI spec = 0.35 UI was not exceeded. The 312.5MHz in x10 mode yields a better result of 0.32UI.

Figure 3. ispClock5400D with LatticeECP3 in XAUI Mode

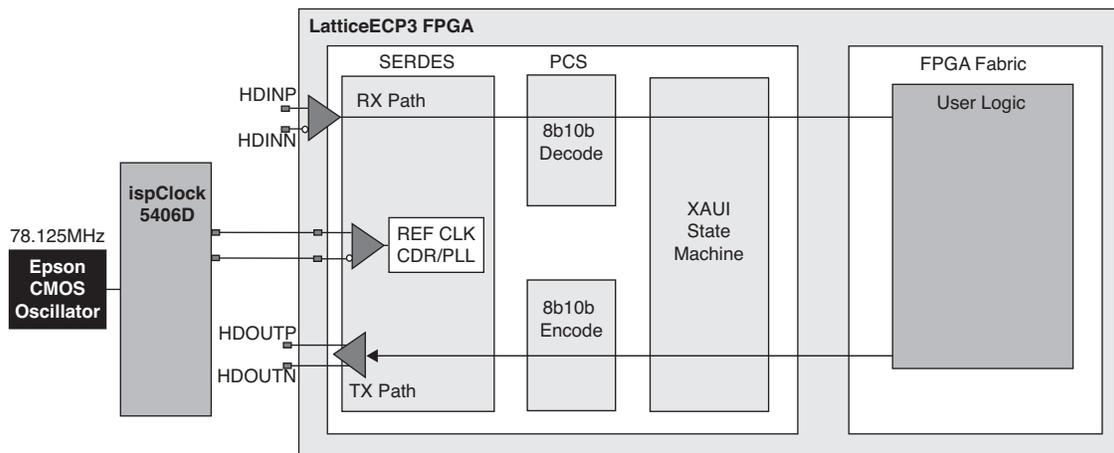


Figure 4. LatticeECP3 Serial Protocol Board and ispClock5400D Evaluation Board with SMA Cables

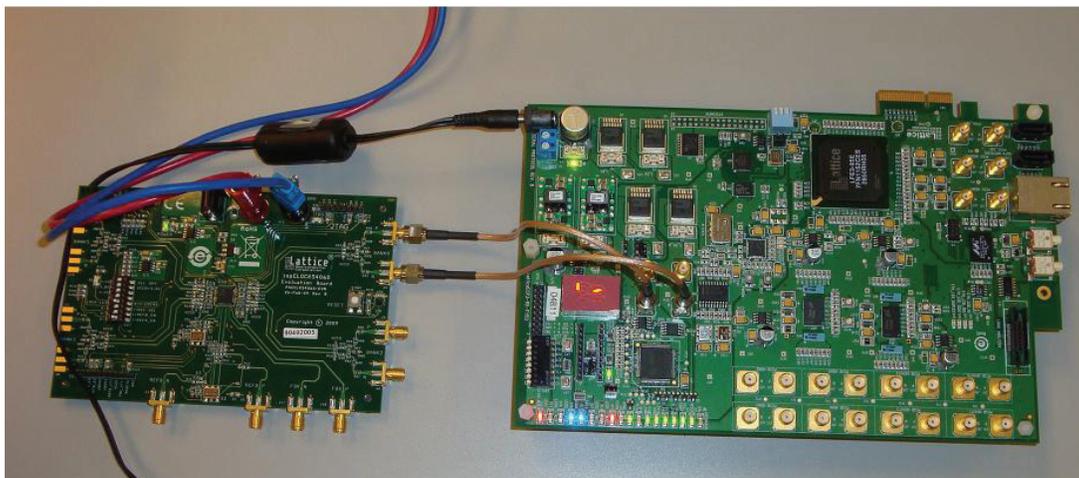


Figure shows the board test set-up that was used for the SERDES protocol tests. The testing was performed using standard Lattice evaluations boards.

The board on the left is the ispClock5400D Evaluation Board and the board on the right is the LatticeECP3 Serial Protocol Board. The boards were interfaced with short SMA cables from the ispClock output bank to the SERDES REFCLK input. Other equipment used included the Agilent 8133 Clock Generator for a reference comparison.

Table 1 summarizes the measured jitter data and calculated UI data. The table compares jitter results from different settings within both the ispClock and in the LatticeECP3 SERDES settings. The ispClock generated two sets of frequencies, shown as X2 and X4. These represent 156.25MHz and 312.5MHz respectively. The SERDES options were X10 mode and X20 mode.

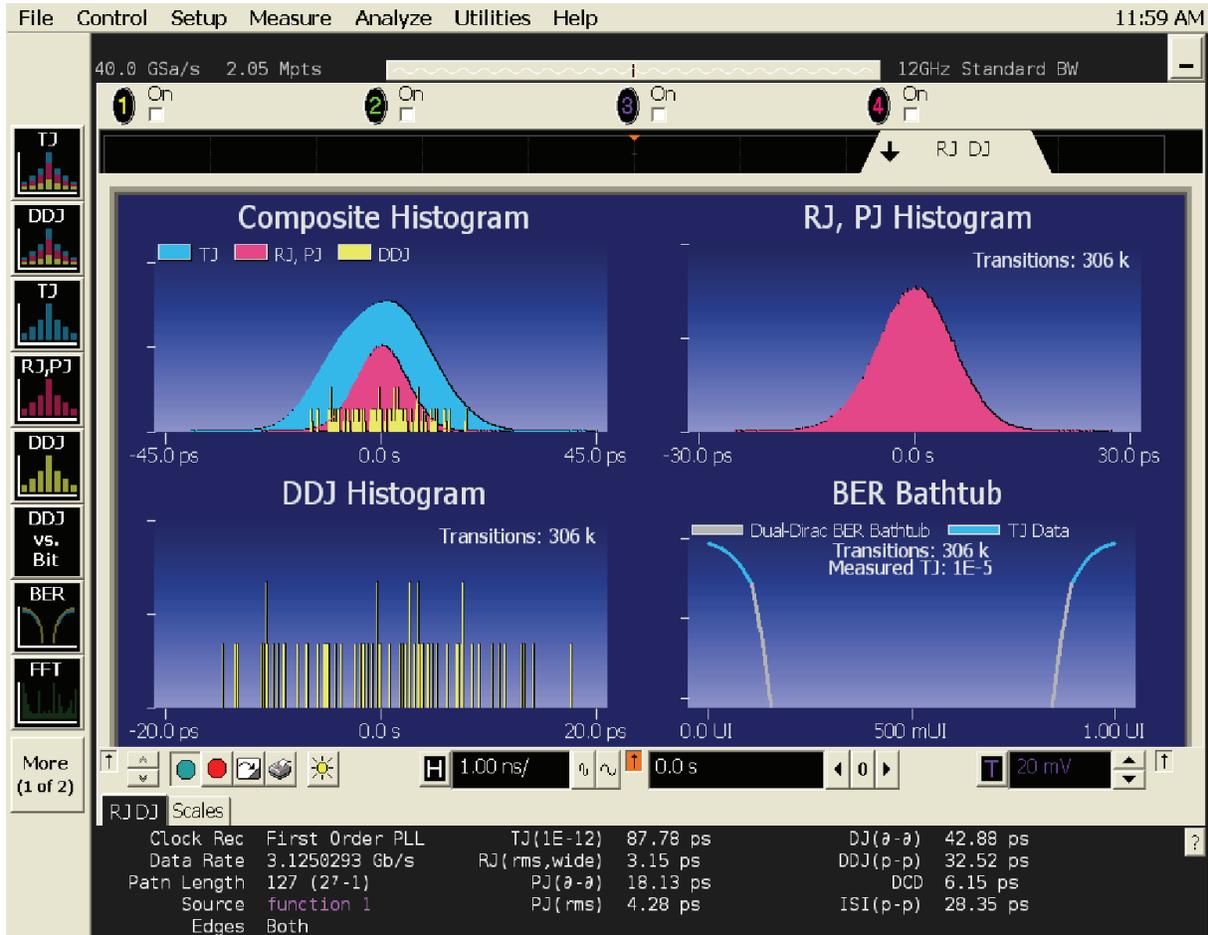
Table 1. Gathered Jitter Data¹

Room Temperature	Tj	Tj-UI	Rj	Rj-UI	Pj	Pj-UI	Dj	Dj-UI
Connor-Winfield 156.25MHz Differential Oscillator	87.8	0.27	3.15	0.0098	18.13	0.0567	42.88	0.1340
Agilent 8133 Clock Generator	87.3	0.27	3.05	0.0095	18.44	0.0576	43.86	0.1371
Epson 78.125MHz Fixed with ispClock5406D X4, 312MHz X10 Mode								
-55°C	105.65	0.33	2.78	0.0087	43.67	0.1365	65.64	0.2051
0°C	101.04	0.32	2.78	0.0087	38.16	0.1193	61.39	0.1918
+85°C	99.44	0.31	2.8	0.0088	37.43	0.1170	59.51	0.1860
Epson 78.125MHz Fixed with ispClock5406 X2, 156.25MHz X20 Mode								
-55°C	112.31	0.35	2.72	0.0085	51.23	0.1601	73.5	0.2297
0°C	110.26	0.34	2.74	0.0086	48.7	0.1522	71.12	0.2223
+85°C	110.74	0.35	2.86	0.0089	47.37	0.1480	69.91	0.2185

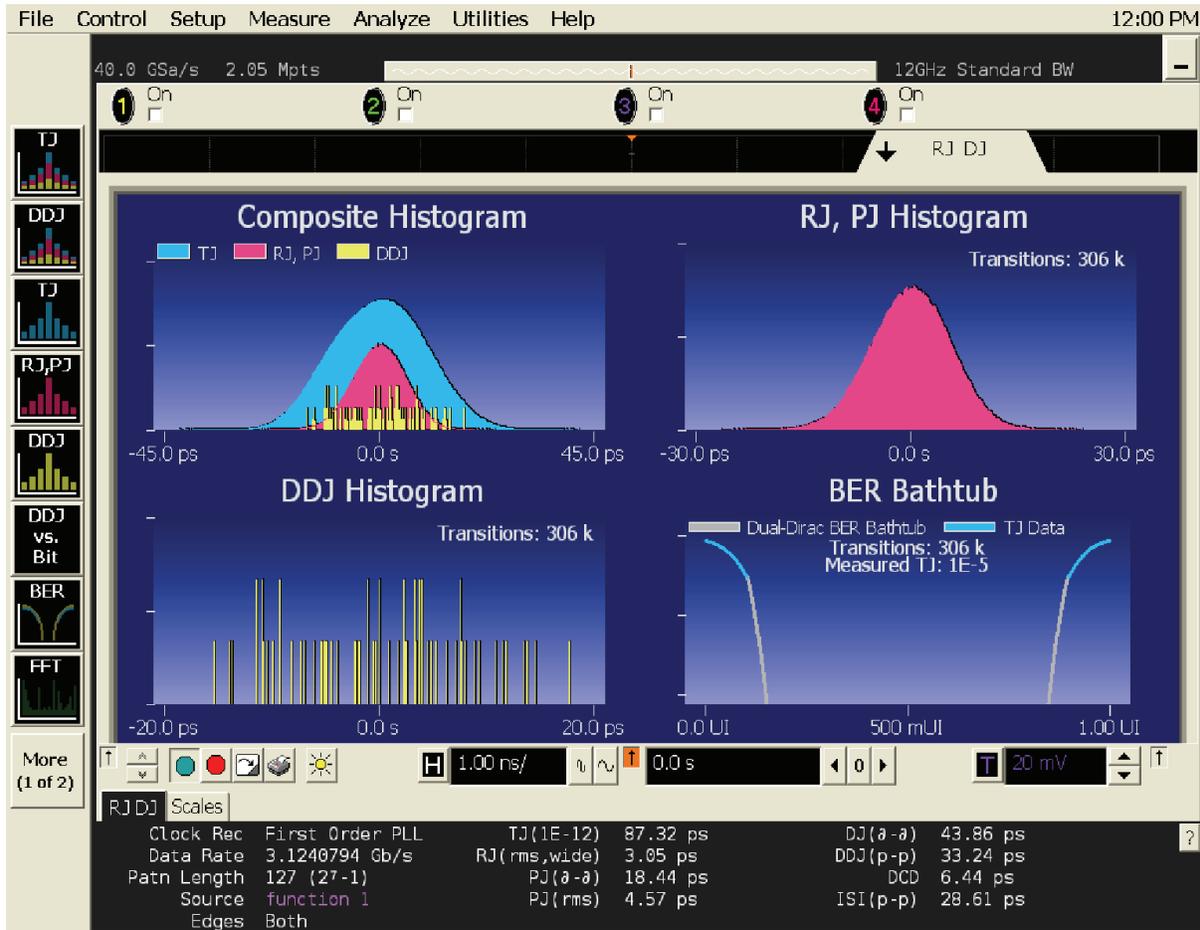
1. The LatticeECP3 SERDES receive and transmit modes had no errors with PRBS7 at -55°C, 0°C, and +85°C across both reference clock combinations with the fixed frequency oscillator and X2, X4 configurations.

For reference, the following plots show histograms of data gathered for the more expensive Conner-Winfield differential oscillator and the Agilent clock generator, which was used as a reference clock source. Note that both plots are represented at the top of Table 1.

Clock Source: Conner-Winfield 156.25MHz Differential Oscillator
Overall Result: Tx Jitter 3.125Gbps PRBS7 Data
Test Design: LatticeECP3 Serial Protocol Board, First Order PLL Filter



Clock Source: Agilent 8133 Clock Generator
Overall Result: Tx Jitter 3.125Gbps PRBS7 Data
Test Design: LatticeECP3 Serial Protocol Board, First Order PLL Filter



SDI Video Application

The ispClock5400D was also tested in a video application along with the LatticeECP3 FPGA. For video systems and cards, it is important that the main clock that feeds the video is stable and low jitter. The SMPTE application programmed into the LatticeECP3 consisted of video control for packetization from parallel to serial. Encryption was handled by the video IP and control logic for manipulating the video data. The frequency of interest for this application is both 27MHz and 270MHz. This solution consists of a 45 MHz Epson CMOS oscillator feeding the ispClock device along with some external divider logic that was used to generate this non-2n frequency. The goal was to provide a low-cost solution that meets the strict requirements of a low-jitter clock. The 270MHz was generated internally with the PLL of the LatticeECP3.

One thing that differs in this application is that some of the logic is borrowed from within the FPGA to build the divider needed for the ispClock. The external divider allows the ispClock to generate the necessary frequencies like the 27MHz for video. The input REFCLK for ispClock5406 was a low-cost Epson oscillator packaged in a 5x7mm ceramic SMD (part number SG-710ECK E45.000Mhz). This, coupled with external feedback divider of 6, generated a very clean 27MHz differential clock.

The ispClock device family has differential outputs and allows either single-ended or differential inputs.

The 27MHz jitter must be very low as it is multiplied up to 270MHz with a PLL in the LatticeECP3, and used to shift video data into the frame. The application was tested with a LatticeECP3 Video Protocol Board. The FPGA held

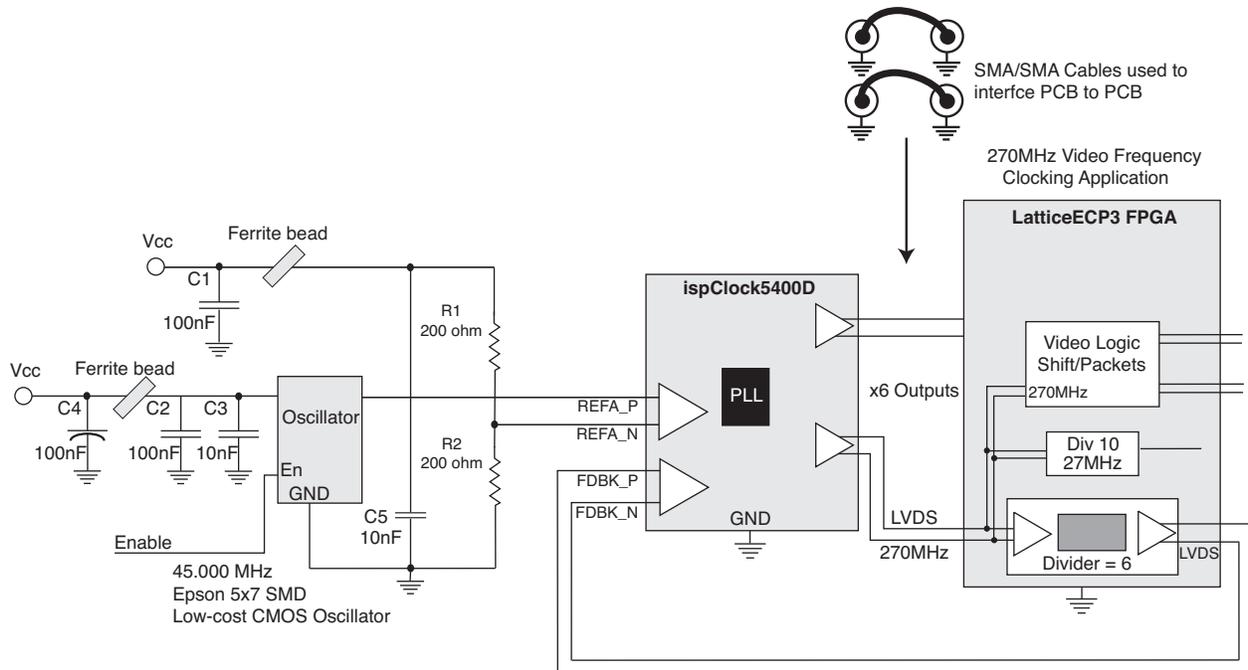
the video IP and the divider for the ispClock, as well as all the logic for sending and receiving and alignment of the video stream pattern. All pattern tests, as well as video in and video out tests, passed the SMPTE SDI standards.

Figure 5 shows the LatticeECP3 Video Protocol Board and the ispClock 5400D Evaluation Board with SMA cables.

Figure 5. Video Application Interface, Epson Oscillator (45 MHz) Used with External Divider



Figure 6. LatticeECP3 Video Application



Conclusion

From the collected data and testing of real world applications we see that the ispClock5400D devices interface well in SERDES-based designs. This application note demonstrates that the ispClock5400D device, along with a low-cost Epson CMOS oscillator, can replace expensive differential oscillators. The ispClock5400D device was driven from the different oscillators and test equipment to show the relative jitter as a complete solution. To further the test coverage, the ispClock device was tested at cold, room and hot temperatures. The results demonstrate the effectiveness of the ispClock5400D as a low-cost SERDES clock source.

Technical Support Assistance

Hotline: 1-800-LATTICE (North America)
+1-503-268-8001 (Outside North America)

e-mail: isppacs@latticesemi.com

Internet: www.latticesemi.com

Revision History

Date	Version	Change Summary
October 2009	01.0	Initial release.